

1 Small tutorial to designpid.m

This pdf-document gives two small examples on how to use the PID optimization software tool called `designpid`. The function has been tested thoroughly in Matlab 7 (R2007a), but may give small errors in other versions. If you have any comments, questions or bug reports regarding the software, I'll be happy to hear from you. You can reach me via the e-mail address `olof.garpinger(at)control.lth.se`, where you have to replace (at) with @.

1.1 Example 1

First assume that we want to find an optimal PID controller for the first order system with time delay

$$P(s) = \frac{1}{s+1}e^{-s},$$

which is a standard model that can easily be derived by for instance a step response test. Let us keep all settings to default values ($M_s = M_p = 1.4$, $T_f = 0.001$ etc.). Now, start out by defining the process in Matlab:

```
>> s = tf('s')
>> P = 1/(s+1);
>> P.iodelay = 1
```

Transfer function:

$$\frac{1}{\exp(-1*s) * (s + 1)}$$

Then run the Matlab function without use of the menu (default run):

```
>> design = designpid(P,0);

Nelder Mead iteration number: 10
Nelder Mead iteration number: 20
Nelder Mead iteration number: 30
Final number of Nelder Mead iterations: 33

PID Parameters:
K-value:    0.6551
Ti-value:   0.9616
Td-value:   0.3979

IAE-value:  1.7312
```

Total time for the algorithm to finish: 16.38 seconds

and we have our optimal PID controller.

1.2 Example 2

Let us now change the settings a bit, but still use the same process, so that we instead specify:

$$\begin{aligned}M_s &= 2.0 \\M_p &= 2.0 \\T_f &= 0.1\end{aligned}$$

and use a PI controller instead of a PID. Then we can go through the menu:

```

>> design = designpid(P,1);
-----
1. Change the optimization settings
2. Change the M-circle settings
3. Set the frequency span
4. Change the low pass filter
5. Graphic settings
6. Choose to design a PI or PID
7. Do not change anything else
Choose what settings to change: 2
-----
M-circle settings (default Ms=Mp=1.4, press return):
Set maximum gain of the sensitivity function, Ms>1: 2
Set maximum gain of the complementary sensitivity function, Mp>1: 2
-----
1. Change the optimization settings
2. Change the M-circle settings
3. Set the frequency span
4. Change the low pass filter
5. Graphic settings
6. Choose to design a PI or PID
7. Do not change anything else
Choose what settings to change: 4
-----
Low pass filter (1/(0.5*(s*Tf)^2+s*Tf+1)) settings (default Tf=0.001),
(1/(Tf*s+1) if PI controller):
Set lowpass filter time constant Tf: 0.1
-----
1. Change the optimization settings
2. Change the M-circle settings
3. Set the frequency span
4. Change the low pass filter
5. Graphic settings
6. Choose to design a PI or PID
7. Do not change anything else
Choose what settings to change: 6
-----
Choose to design a PI or PID controller (default N):
Do you wish to design a PI controller (Y/N)?: Y
-----
1. Change the optimization settings
2. Change the M-circle settings
3. Set the frequency span
4. Change the low pass filter
5. Graphic settings
6. Choose to design a PI or PID
7. Do not change anything else
Choose what settings to change: 7

Nelder Mead iteration number: 10
Final number of Nelder Mead iterations: 10

```

PID Parameters:

K-value: 0.7742

Ti-value: 1.3254

Td-value: 0.0000

IAE-value: 1.7517

Total time for the algorithm to finish: 6.86 seconds

The PID parameters can be found in the `design` struct under `design.PID.Kfinal`, `design.PID.Tifinal` and `design.PID.Tdfinal`.

```
>> design.PID
```

```
ans =
```

```
      Tf: 0.1000  
Lpfilter: [1x1 tf]  
      Tiamigo: 1.0464  
      IAEmin: 1.7517  
      Kfinal: 0.7742  
      Tifinal: 1.3254  
      Tdfinal: 0  
      Grfinal: [1x1 tf]
```

Good luck and hope you find the software tool useful!

Sincerely,

Olof Garpinger
Lund, Sweden
30/6-2008